CLAIMS

We claim:

1. A method of managing communications between service components in a computing environment, the computing environment comprising a plurality of interconnected processing nodes, each of the service components having a respective identity and being programmed on at least a respective one of the processing nodes, the method comprising:

based on the identity of at least one of the service components, establishing access-control logic restricting inter-node communication involving the at least one service component; and

applying the access-control logic to block an inter-node communication involving the at least one service component.

2. The method of claim 1, wherein establishing the access-control logic comprises:

establishing a rule indicating whether to allow a communication involving the at least one service component; and

translating the rule into the access-control logic.

3. The method of claim 2, wherein:

establishing a rule indicating whether to allow a communication involving the at least one service component comprises establishing a rule indicating whether to allow a communication with a first service component programmed on a first processing node in the computing environment.

4. The method of claim 3, wherein:

establishing a rule indicating whether to allow a communication involving the at least one service component comprises establishing a rule indicating whether to allow a communication between (i) a first service component programmed on a first processing node in the computing environment and (ii) a second service component programmed on a second processing node in the computing environment.

5. The method of claim 3, wherein the at least one service component resides at at least one service-access-point in the computing environment, and wherein translating the rule into the access-control logic comprises mapping the rule into packet-filter logic keyed to the at least one service-access-point.

6. The method of claim 5, wherein the at least one service-access-point comprises an IP address of the first processing node.

7. The method of claim 6, wherein the first processing node is programmed to associate a first transport port with the first service component, and wherein the at least one service-access-point further comprises the first transport port.

8. The method of claim 1, wherein applying the access-control logic to block an inter-node communication involving the at least one service component comprises:

detecting an attempted inter-node communication involving the at least one service component;

based on the access-control logic, making a determination that the attempted inter-node communication should be blocked; and

in response to the determination, blocking the attempted inter-node communication.

9. The method of claim 1, wherein at least two processing nodes of the plurality of interconnected processing nodes run different operating systems.

10. The method of claim 1, wherein at least two processing nodes of the plurality of interconnected processing nodes support different processor instructions sets.

11. The method of claim 1, wherein the computing environment is a cluster-based computing environment.

12. The method of claim 1, wherein the computing environment is a public computing platform.

13. A method of managing communications between service components in a computing environment, the computing environment comprising a plurality of interconnected processing nodes, each of the service components having a respective identity and being programmed on at least a respective one of the processing nodes, the method comprising:

based on the identity of at least one of the service components, establishing at least one access-control rule indicating whether to allow at least one communication involving the at least one service component;

translating the at least one access-control rule into access-control logic;

detecting an attempted inter-node communication between service components;

based on the access-control logic, determining that the attempted inter-node communication between service components is not allowed; and

responsively blocking the attempted inter-node communication.

14. The method of claim 12, wherein the access-control logic comprises packet-filter logic.

15. The method of claim 12, wherein:

each of the service components is designated by a respective service-access-point (SAP) in the computing environment, and each processing node has a respective SAP as well; and

the access-control logic comprises packet-filter logic keyed to at least one SAP in the computing environment.

16. The method of claim 15, wherein the respective SAP of each service component comprises an IP address of the respective processing node on which the service component is programmed, and wherein the access-control logic comprises packet-filter logic keyed to at least one such IP address.

17. The method of claim 16, wherein at least one of the SAPs of a service component further comprises a port selected from the group consisting of a TCP port and a UDP port, and wherein the packet-filter logic is further keyed to at least one such port.

18. The method of claim 13,

wherein the communications between service components are packet-based; and

wherein the access-control logic comprises packet-filter logic keyed to a combination of at least (i) a packet transport protocol, (ii) a source address in the computing environment and (iii) a destination address in the computing environment.

19. The method of claim 13, wherein translating the at least one access-control rule into access-control logic comprises:

mapping the at least one access-control rule to packet-filter logic keyed to at least one service-access-point in the computing environment.

20. The method of claim 13,

wherein at least a given one of the processing nodes includes a firewall for restricting communications with the given processing node; and

translating the at least one access-control rule into access-control logic comprises provisioning the firewall of the given processing node to allow communications between at least one service component programmed on the given processing node and at least one service component programmed on another processing node.

21. The method of claim 13, wherein the processing nodes are interconnected by an interconnection system, and wherein the interconnection system performs at least the elements of (i) detecting an attempted inter-node communication and (ii) blocking the attempted inter-node communication.

22. The method of claim 21, wherein the interconnection system further performs the element of determining that the attempted inter-node communication is not allowed.

23. The method of claim 21, further comprising:

provisioning the interconnection system with the access-control logic.

24. The method of claim 23, wherein provisioning the interconnection system with the access-control logic comprises pre-provisioning the interconnection system.

25. The method of claim 23, wherein provisioning the interconnection system with the access-control logic comprises provisioning the interconnection system on demand.

26. The method of claim 23, wherein the interconnection system comprises a switch, and wherein provisioning the interconnection system with the access-control logic comprises setting up the switch to apply the access-control logic.

27. The method of claim 26,

wherein the switch comprises (i) a packet-filtering agent and (i) a provisioning-interface for receiving instructions to set up the packet-filtering agent, the switch being arranged to translate the instructions into packet-filtering logic executable by the packet-filtering agent; and

wherein setting up the switch to apply the access-control logic comprises providing the switch, via the provisioning-interface, with instructions representative of the access-control logic.

28. The method of claim 21, wherein an entity external to the interconnection system performs the element of determining that the attempted inter-node communication is not allowed.

29. The method of claim 28, further comprising:

the external entity instructing the interconnection system to block the communication, in response to a determination that the attempted inter-node communication is not allowed.

30. The method of claim 28, wherein the external entity comprises a session manager.

31. The method of claim 21, wherein the interconnection system comprises a switch.

32. The method of claim 21, wherein the interconnection system comprises a router.

33. The method of claim 13, wherein the computing environment is a cluster-based computing environment.

34. The method of claim 13, wherein the computing environment is a public computing platform operated by a platform-provider.

35. The method of claim 13, wherein the attempted inter-node communication comprises an attempted inter-node communication between antagonistic service components.

36. The method of claim 13, wherein the attempted inter-node communication comprises an attempted communication of a packet from a first processing node to a second processing node, and wherein blocking the attempted communication comprises dropping the packet.

37. A method for managing application logic in a public computing platform, the public computing platform comprising a network of processing nodes interconnected by an interconnection system, the method comprising:

receiving specifications of at least two computer-program applications, the applications cooperatively comprising a number of application components;

loading the application components of the at least two applications onto at least two of the processing nodes of the computing platform; and

applying access-control rules defining allowed communications between the application components, to restrict inter-node communication between the application components.

38. A computing environment with communication control comprising:

an interconnection system;

a plurality of co-located processing nodes interconnected via the interconnection system;

a plurality of application components loaded onto the processing nodes, each application component having a respective service-access-point defining location of the application component in the computing environment;

logic indicating allowed inter-node communications between application components;

the logic being executable, in response to an attempted inter-node communication, to make a determination of whether the attempted inter-node communication is allowed; and

the logic being executable, in response to a determination that the attempted inter-node communication is not allowed, to block the attempted inter-node communication.

39. The computing environment of claim 38, wherein the logic is located, at least in part, external to the interconnection system.

40. The computing environment of claim 39, wherein the logic is located, at least in part, in a session manager communicatively linked with the interconnection system.

41. The computing environment of claim 40, wherein the session manager provides at least a portion of the logic to the interconnection system on demand.

42. The computing environment of claim 38, wherein the computing environment is a cluster-based computing environment.

43. The computing environment of claim 38, wherein the computing environment is a public-computing platform.

44. The method of claim 38, wherein at least two of the co-located processing nodes run different operating systems.

45. The method of claim 38, wherein at least two of the co-located processing nodes support different processor instructions sets.